CS231 Digital Logic Design

Week #04

1. Express the following numbers in decimal (Show Your Steps)
   a. $(10110.0101)_2$
   22.3125
   b. $(16.5)_{16}$
   22.3125
   c. $(26.24)_8$
   22.3125
   d. $(DADA.B)_{16}$
   56026.6875
   e. $(1010.1101)_2$
   10.8125

2. Perform subtraction on the given unsigned binary numbers using the 2's complement of the subtrahend. Where the result should be negative, find its 2's complement and affix a minus sign.
   a. 10011−10010 = 00001
   b. 100010−100110 = -(000100)
   c. 1001−110101 = -(101100)
   d. 101000−10101 = 10011
3. Simplify the following Boolean expressions to a minimum
   a. xy + xy` = x
   b. (x + y)(x + y`) = x
   c. xyz + x`y + xyz` = y

   d. (A + B)`(A` + B`)` = 0

   e. (a + b + c`)(a`b` + c) = ac + bc + c`a`b`

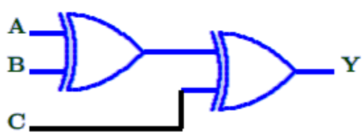
   f. a`bc + abc` + abc+ a`bc` = b

CS231 Digital Logic Design

Week #05

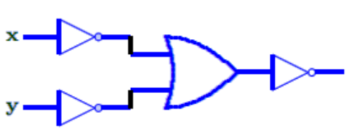
1. Draw the logic diagrams of the following circuits using CircuitVerse and establish their truth tables.

a)



A	B	C	Y
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

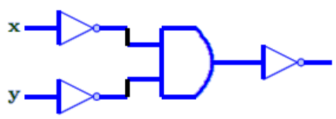
b)



X	Y	F
0	0	0
1	0	0
0	1	0
1	1	1

CS231 Digital Logic Design

c)



X	Y	F
0	0	0
1	0	1
0	1	1
1	1	1

CS231 Digital Logic Design

2. Simplify the following Boolean expressions to a minimum :

   a) (A+B')(AB'+C)C = CA + CB`

   b) (A+B')C+(AB'+C).C = C

   c) ((AB)'+(CD)')' = ABCD

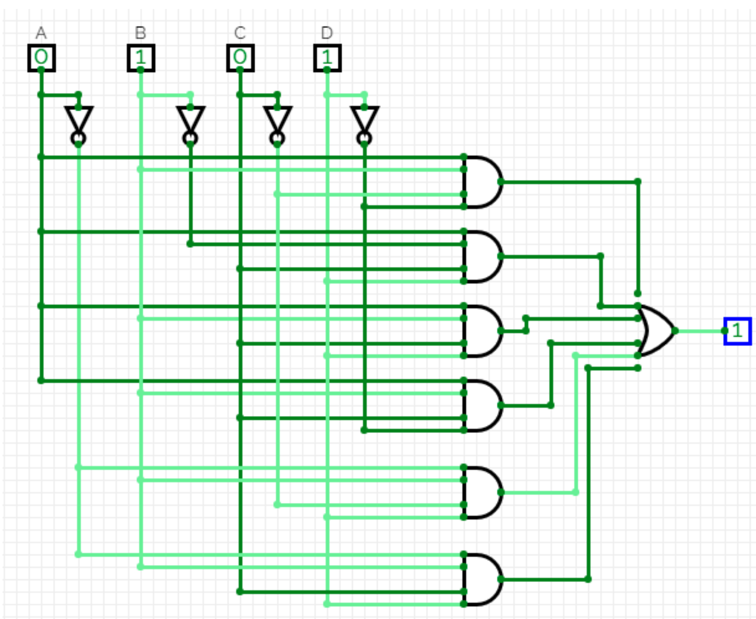
3. Use the Karnaugh table to simplify the following logic equation:

   X=ABC'D'+AB'CD+ABCD+ABCD'+A'BC'D+A'BCD

   Draw its logic diagram circuit using CircuitVerse before and after simplification

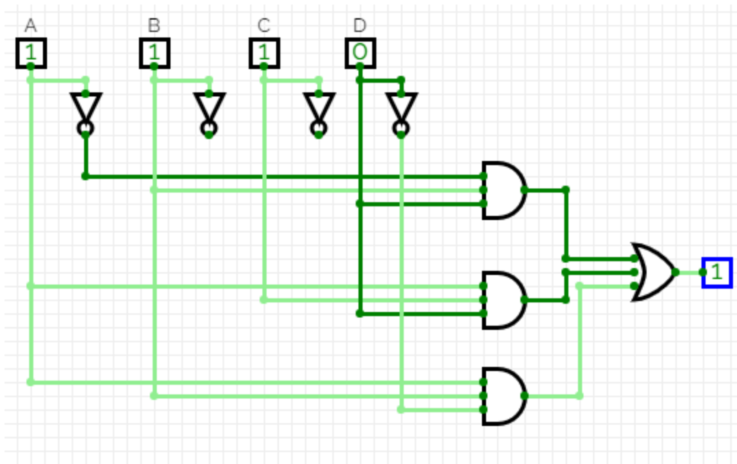
   X = A`BD + ACD + ABD`

   Original circuit



CS231 Digital Logic Design

Simplified circuit



CS231 Digital Logic Design

Week #06 Lab

1. Minimize the following Boolean function using K-map

   F(A, B, C, D) = Σm(0, 1, 2, 5, 7, 8, 9, 10, 13, 15)
   F (A, B, C, D) = BD + C'D + B'D'

2. Minimize the following Boolean function:

   F(A, B, C) = Σm(0, 1, 6, 7) + Σd(3, 4, 5)
   F = A + B'

3. Minimize the following Boolean function and represent it as
   a. SOP
   b. POS

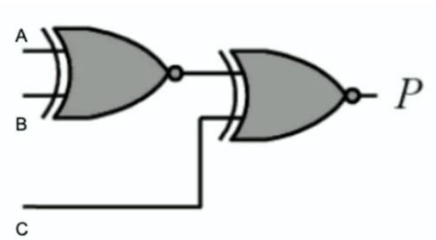
   F(A, B, C, D) = Σm(0, 2, 8, 10, 14) + Σd(5, 15)

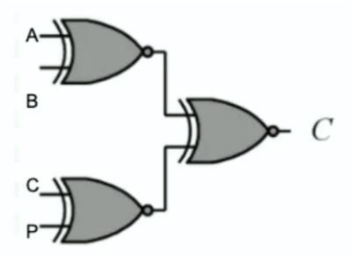
   SOP
   F = A + BC

   POS
   F' = A'(B' + C')

CS231 Digital Logic Design

4. Derive the circuits for a three-bit parity and four-bit parity generator using an odd parity bit.





CS231 Digital Logic Design

Week #07 Lab

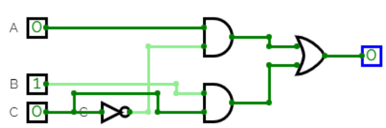
1. Solve the following questions:

   A. Derive the function from the truth table.
   B. Simplify the resulted function and draw the logical diagram.
   C. Finally, convert the diagram to All NAND gates

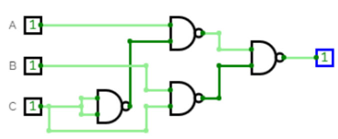
C	B	A	Y
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

Y= C’A+CB

Logical diagram after simplification

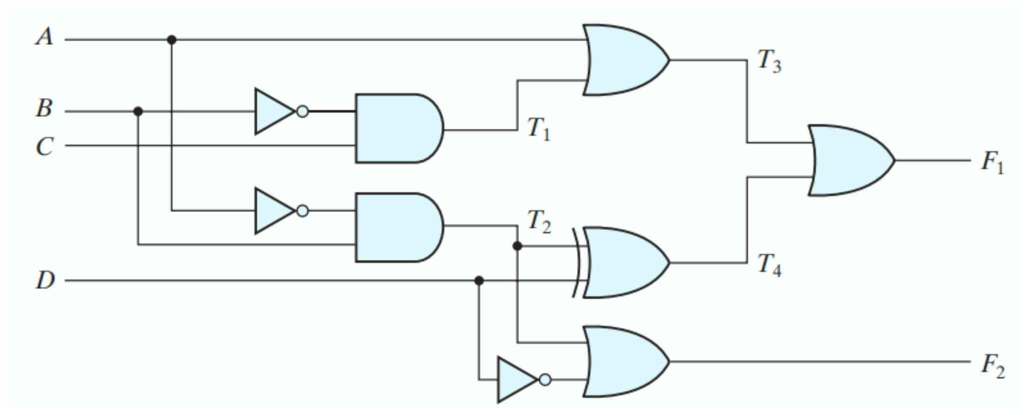


Logical diagram after converting all to NAND gates



CS231 Digital Logic Design

2. Consider the combinational circuit



A. Derive the Boolean expressions for T1 through T4. Evaluate the outputs F1 and F2 as a function of the four inputs.
B. List the truth table with 16 binary combinations of the four input variables. Then list the binary values for T1 through T4 and outputs F1 and F2 in the table.
C. Plot the output Boolean functions obtained in part (b) on maps and show that the simplified Boolean expressions are equivalent to the ones obtained in part (a).

F1 = A + B'C + BD' + B'D
F2 = A'B + D'

CS231 Digital Logic Design

3. Design a combinational circuit with three inputs and one output. (a)* The output is 1 when the binary value of the inputs is less than 3. The output is 0 otherwise. (b) The output is 1 when the binary value of the inputs is an even number.

You need to follow the design process :
a) naming variables.
b) deriving the truth table based on the relationship between the variables.
c) simplifying the expression of the function by using k-map (or other methods).
d) drawing the logic diagram.
e) test with different inputs from the truth table and compare the output you got from the logic diagram